Armado de un microcomputador, basado en microcontrolador 8031

Objetivos.

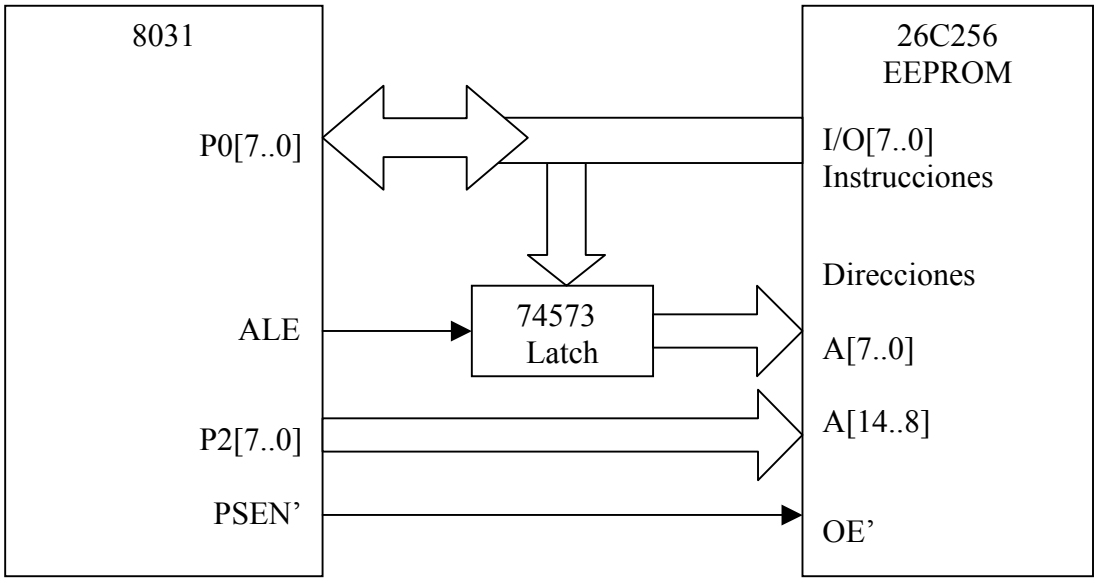
- Conocer la forma en que están interconectadas la memoria, el controlador, y el CPLD, en la tarjeta de desarrollo. Entender con detalles el diseño lógico, en el CPLD, que permite ejecutar programas en en controlador 8031.
- Aprender a grabar programas hexadecimales con el código del programa compilado en la memoria RAM de la tarjeta. Grabar el mapa de bits en el CPLD.
- Ejecutar programas simples, que permitan visualizar las señales de control: PSEN, ALE, RD, WR mediante un osciloscopio.
- Observar un ciclo de lectura de la instrucción. Ciclo de búsqueda(fetch).
- Observar la ejecución de una instrucción movx.

Preparación previa.

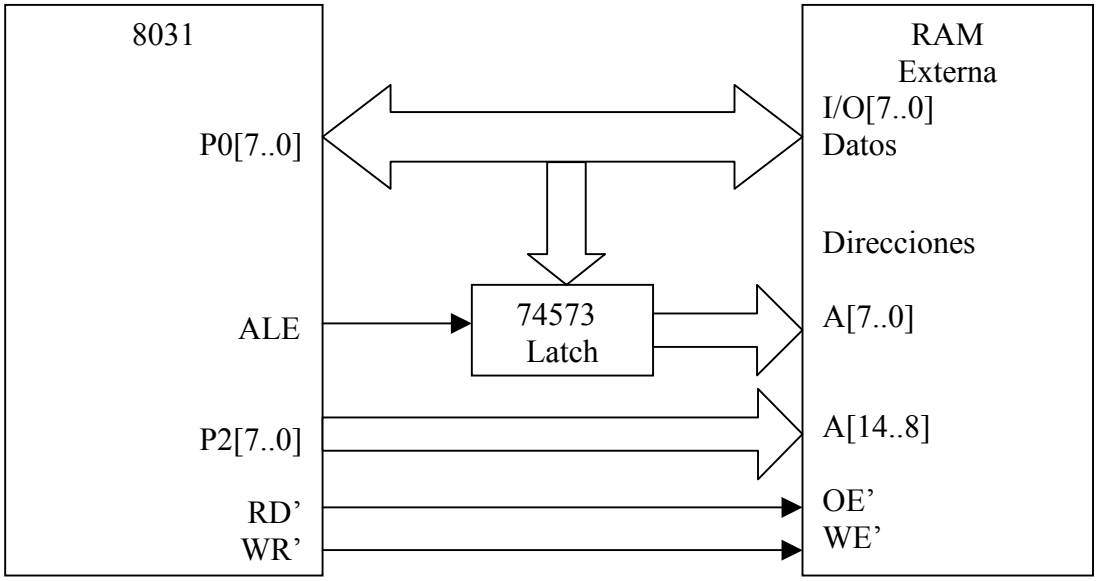
a) Conexión de la memoria de programa.

Estudiar en el User manual: página 1-7 y figura 36 en pág. 3-34.

Determinar las razones: para emplear el latch controlado por ALE, y las conexiones de PSEN' a OE´.



b) Conexión de la memoria RAM. Compartir bus de datos y direcciones del caso a).
Estudiar en el User manual: página 1-8 y figura 37 y 38 en págs. 3-34 y 3-35.
Razones de las conexiones RD y WR hacia OE y WE.



c) Conexión de circuito de power-on reset. Fig. 26 en pág. 3-27 Agregar un reset manual libre de rebotes.
d) Conectar el cristal y condensadores. Ver fig 30 en pág. 3-31.
En los esquemas mostrados se supone memoria de datos(RAM) y de programa(EEPROM) separadas. Veremos a continuación el esquema de conexiones para memoria común de datos y programa empleando la tarjeta XS95.

Descripción de la lógica para el funcionamiento del µC8031 en la tarjeta XS95.

Primero describiremos la lógica necesaria para el funcionamiento del microcontrolador dentro de la Tarjeta XS95.

Observando el diagrama de conexiones entre los dispositivos de la tarjeta XS95 presentado en la figura 4, se puede apreciar lo siguiente:

- Puerto 2 del µC8031 (P2.7.. P2.0) se encuentra permanentemente conectado a los respectivos pines de la parte alta del bus de direcciones de la memoria SRAM (A15..A7). Además este puerto está disponible en el CPLD(pines 34, 61, 57, 53, 55, 54, 56, 58). Si se desea escribir en la RAM mediante un programa en el CPLD, estos pines deben ser programados como salida. Pero si el 8031 lee o escribe en la memoria de datos o de programa, estos pines deben ser programados de entrada en el CPLD.
- Puerto 0 (P0.7..P0.0) del µC8031 se encuentra permanentemente conectado al bus de datos de la memoria SRAM(D7..D0). Además este bus entra al CPLD en los pines 35, 36, 37, 39, 40, 41, 43, 44. Estos pines deben ser programados como entradas a un registro latch en el CPLD; el control de escritura de ese registro, que sostiene la parte baja de la dirección, está gobernado por la señal ALE. La señal ALE debe ser programada como entrada, en el pin 20 del CPLD.

- Señal de habilitación de escritura en memoria de datos externa del µC8031 (P3.6) se encuentra permanentemente conectada al pin WE' de la memoria SRAM. Esta señal también se ingresa al CPLD, a través del pin 63 que debe ser usado sólo como de entrada al CPLD.
- Señal RD' de habilitación de lectura de memoria externa debe ser programada como pin de entrada al CPLD, en pin 32.
- Señal PSEN' de habilitación de lectura de memoria de programa debe ser programada como pin de entrada al CPLD, en pin 13.
- Señal OE' de habilitación de salida de memoria estática debe ser programada como pin de salida del CPLD, en pin 62. Dentro del CPLD, puede programarse un and de las señales RD' y PSEN', que salga por el pin 62, que gobierne los instantes en que el 8031 lea el puerto P0, para ingresar el byte que está saliendo por el bus de datos de la memoria RAM(que trae un byte de instrucción, cuando PSEN' está baja; o un byte de datos cuando RD' está baja) . El procesador no puede bajar simultáneamente RD' y PSEN'. Nótese que el and, efectúa el or de las señales de lógica negativa.
- Señal RST para resetear el 8031. Al subir a nivel alto, por lo menos 24 ciclos de reloj, pone al procesador en un estado inicial(ver tabla 2 en pág. 2-9 del manual). Deja a PC con valor 0x0000, las puertas en 0xFF, al SP en 0x07. Al deshabilitar, bajando a cero la señal RST, el procesador comienza a ejecutar la instrucción ubicada en la dirección 0x0000. La señal RST debe ser programada como pin de salida del CPLD, en el pin 45. Puede programarse uno de los pines de entrada al CPLD, que traen los bits de la puerta paralela, de tal modo que salga por el pin 45. De esta manera puede lograrse el reset del 8031, con el programa GXSPORT.
- El pin 10 del CPLD se programa como salida, y debe ser la señal de reloj del procesador. El pin 21 del CPLD, que sólo puede programarse como de entrada, trae la señal del reloj programable, puede programarse para que salga(dividida en frecuencia si es necesario) por el pin 10 del CPLD.
- Los pines 26 y 33 del CPLD deben ser programados como salidas, y permiten ingresar eventos ha ser contados por los timer(usados como contadores) del 8031.
- Los pines 68 y 69 del CPLD deben ser programados como salidas, y permiten ingresar eventos que producen interrupciones en el 8031.
- El pin 70 sólo puede ser programado como entrada al CPLD, en caso de emplear la puerta serial del 8031, para transmitir datos en forma serial.
- El pin 31 sólo puede ser programado como salida del CPLD, en caso de emplear la puerta serial del 8031, para recibir datos seriales.
- Los pines A16 y A15 del bus de direcciones de la RAM no existen en la RAM de 32 KB que tiene la tarjeta. Para direccionar 32K direcciones se requieren 15 bits; es decir, están conectados desde A0 hasta A14. Entonces el pin 74 del CPLD, puede considerarse un pin libre, y puede ser programado de entrada o salida. La señal A15 del bus de direcciones del 8031(P2.7) está permanentemenete conectada al pin 35 del CPLD, y si se programa como entrada(del CPLD) y se la dirige a través del CPLD, programando el pin 65(del CPLD) como de salida puede emplearse para deshabilitar la RAM para direcciones mayores de 0x7FFF. En este espacio de direcciones A15 está alta y al ser dirigida a CE' de la memoria RAM

Además, como se cuenta con una única memoria de 32 Kbytes en la tarjeta XS95. En ella deberán coexistir la memoria de programas y la memoria de datos necesarias para el funcionamiento del µC.

La configuración para este caso se muestra en la figura 1.
La conexión del pin EA, puede ser manejada con el uso de un jumper (j7), como se aprecia en la figura 3. Sobre la señal EA ver Accessing external memory en Manual del Usuario página 3-9.

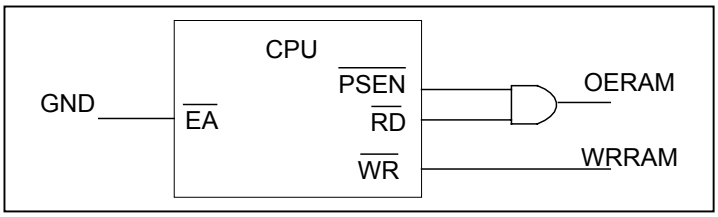


Figura 1: ”Configuración para las señales PSEN, RD y WR del µC8031 en el caso en que deban coexistir las memorias de programas y de datos en un mismo dispositivo de memoria externo”[1]

Tomando en cuenta las observaciones anteriores, la descripción de la lógica para el funcionamiento del µC8031 se simplifica.

La figura 2 muestra un esquema de conexiones general para el funcionamiento del µC8031, en donde las conexiones dibujadas con línea entrecortada representan conexiones internas presentes en la tarjeta XS95, y que por lo tanto no son necesarias de implementar.

Observaciones Adicionales

- Para la habilitación y deshabilitación de la memoria SRAM, se utilizará el bit más significativo del puerto 2, P2.7, del µC8031 (pin 34 CPLD).
  La razón de esta elección es la siguiente: recordando que el µC8031 puede direccionar una memoria de hasta 64KB, y observando el hecho de que la memoria SRAM disponible en la tarjeta XS95 es de 32KB, se advierte que el bit más significativo del bus de direcciones de uC queda libre. Entonces, se utilizará este bit (A15) para habilitar y deshabilitar la memoria SRAM, de manera tal que el rango de direcciones comprendido entre 0x0000 hasta 0x7FFF (32 KB) corresponde a direcciones de la memoria SRAM, y direcciones superiores a este rango, 0x8000 – 0xFFFF, se pueden utilizar para mapear dispositivos externos, ya que en esta zona el bit A15 del bus de direcciones inhabilitará la memoria SRAM, de esta manera el uC puede acceder a estos dispositivos externos realizando operaciones de transferencia de datos (lectura/escritura).

- Para producir el reset del µC8031, se utilizará el bit más significativo del puerto paralelo del PC (pin 9), de esta manera, se podrá resetear desde el PC mediante software.

[1] Gonzáles Vásquez, “Introducción a los microcontroladores”, pág.16.

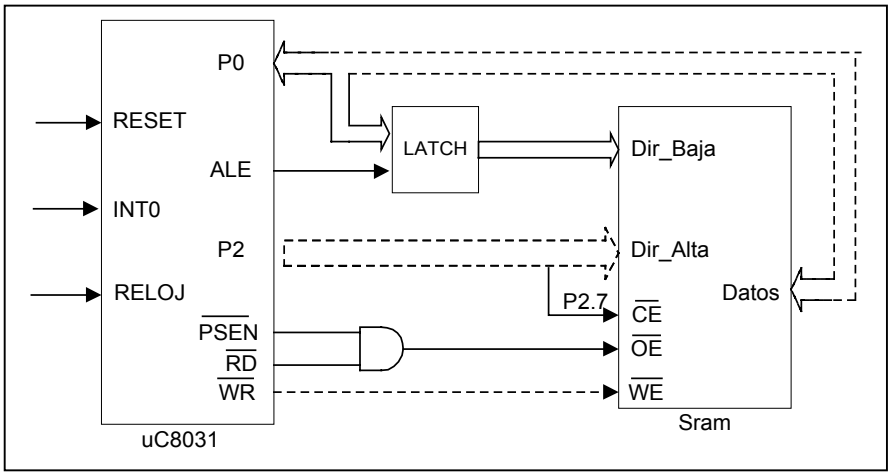


Figura 2: "Esquema general de conexiones para el funcionamiento del µC8031"

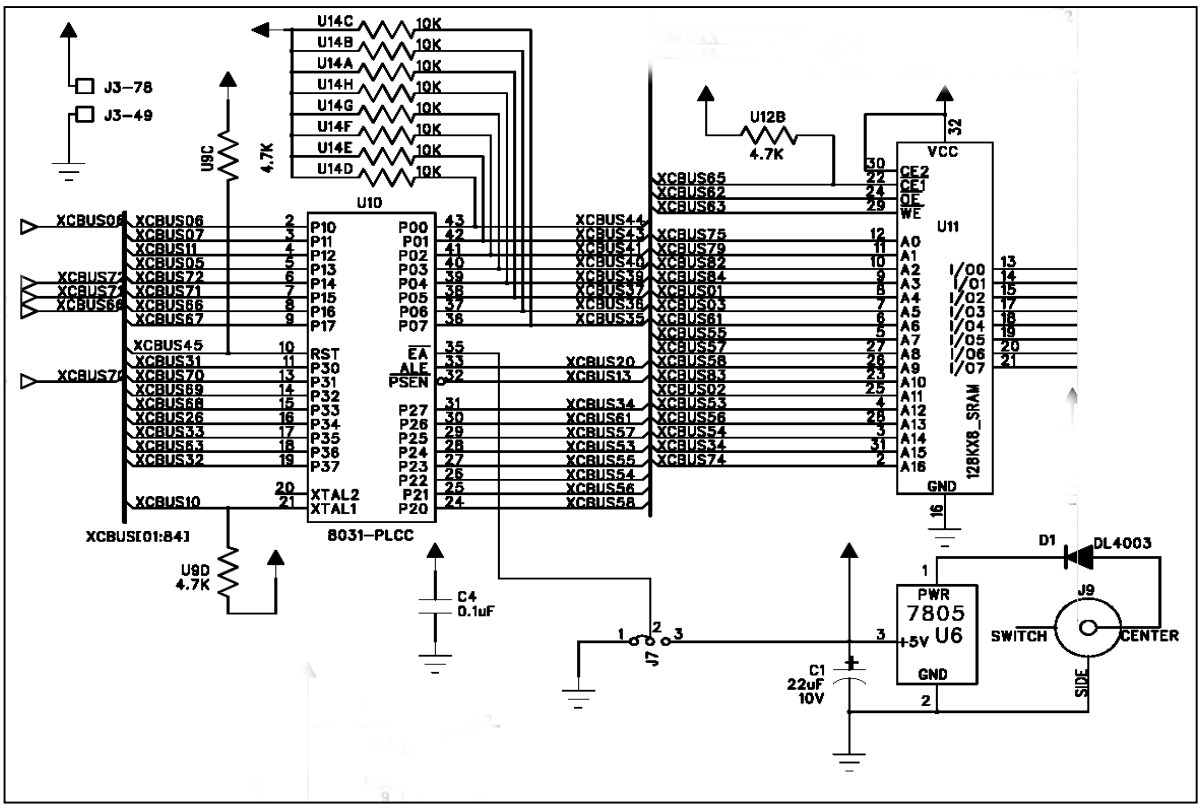


Figura 3: "Detalle de diagrama esquemático para tarjeta XS95, mostrando las conexiones entre el µC8031 y la memoria SRAM. Señal EA."

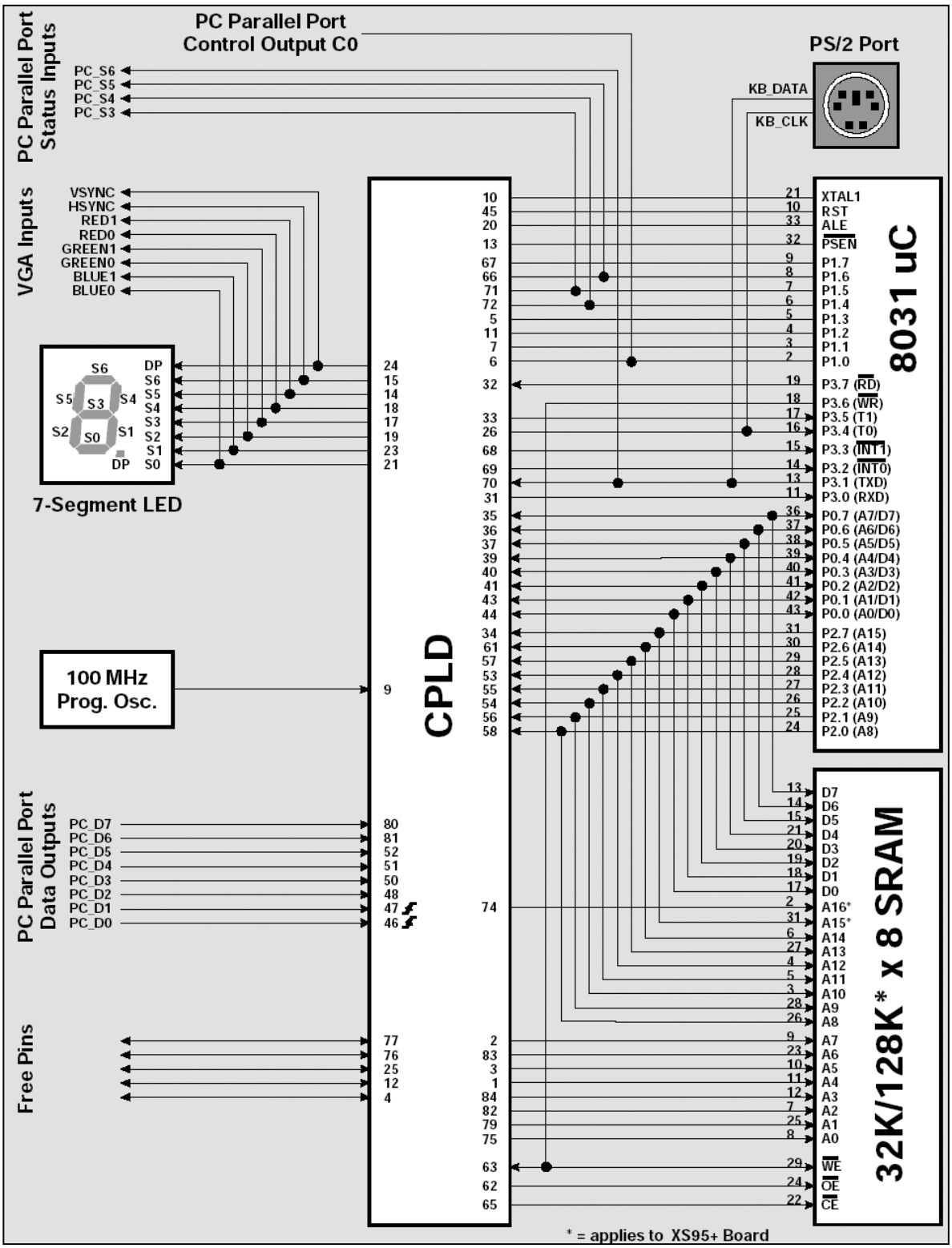


Figura 4: “Diagrama de conexiones entre dispositivos en la tarjeta XS95”

El código ABEL que describe la lógica para el funcionamiento del µC8031, basándose en las figuras 2 y 4 es el siguiente:

```
001 - MODULE uc8031
002 - TITLE 'Módulo que implementa la lógica para el funcionamiento del uC8031'

003 - DECLARATIONS

004 - A15,A7..A0          PIN 34,35,36,37,39,40,41,43,44;
005 - PSEN        PIN 13;
006 - RD          PIN 32;
007 - WR           PIN 63;
008 - CLK         PIN 9;    //
009 - ALE         PIN 20;
010 -
011 - D7..D0      PIN 2,83,3,1,84,82,79,75 istype 'reg';
012 - OE          PIN 62 ISTYPE 'COM'; // señal habilitación de salidas SRAM
013 - CE          PIN 65 ISTYPE 'COM'; // señal habilitación de chip SRAM
014 - RESET       PIN 45 ISTYPE 'COM'; // señal de reset del uC8031
015 - RELOJ       PIN 10 ISTYPE 'COM'; // señal de entrada de clk del uC8031
016 -
017 – PAR7        PIN 80;  // Bit más significativo del puerto paralelo del PC
018 - Bus_dir_baja_sram = [D7..D0];
019 – P0_uc8031  = [A7..A0];

020 - EQUATIONS

021 - Bus_dir_baja_sram.CLK = ! ALE;
022 – Bus_dir_baja_sram := P0_uc8031;
023 - RELOJ = CLK;
024 - CE = A15;
025 - OE = PSEN & RD;
026 – RESET = PAR7;     // bit más significativo de la puerta paralela resetea al uC
027 –
028 – END uC8031          //---- fin descripción para lógica del uC8031
```

"Código en lenguaje ABEL para la descripción de la lógica necesaria para el funcionamiento del µC8031 en la Tarjeta XS95"

El programa debe ser compilado y depositado en el CPLD, para que el microcontrolador tenga conexiones con: la memoria de programa y de datos, el reloj, la señal reset externa.

Adicionalmente pueden agregarse las ecuaciones lógicas que se deseen en el CPLD.

El detalle del listado para este módulo es el siguiente:

Linea1..2: Declaración del nombre y título para el módulo µC8031.
Línea 4: Declaración de las señales necesarias para el bus de direcciones del µC8031, se denominan con los nombres A15..A0, indicando también el pin de la tarjeta XS95 correspondiente a cada uno de ellos.
Línea 5: Declaración de la señal utilizada para la lectura de la memoria de programa del µC8031, denominada con el nombre de PSEN, indicando también el pin de la tarjeta XS95 correspondiente.
Línea 6: Declaración de la señal utilizada para la habilitar la lectura de la memoria de datos del µC8031, denominada con el nombre de RD, indicando también el pin de la tarjeta XS95 correspondiente.
Línea 7: Declaración de la señal utilizada para la escritura de la memoria de datos, denominada con el nombre de WR, indicando también el pin de la tarjeta XS95 correspondiente.
Línea 8: Declaración de la señal proveniente del oscilador programable presente en la tarjeta XS95, denominada con el nombre de CLK, indicando también el pin correspondiente.
Línea 9: Declaración de la señal utilizada para capturar la parte baja del bus de direcciones del µC8031, denominada con el nombre de ALE, indicando también el pin de la tarjeta XS95 correspondiente.
Línea 10:
Línea 11: Declaración de las señales encargadas de contener la parte baja del bus de direcciones del µC8031, se denominan con los nombres D7..D0 y se definen como de tipo registro, de decir, cada una de estas señales corresponde a la salida de un flip-flop de tipo D. Se indican también los pines de la tarjeta XS95 correspondientes.
Línea 12: Declaración de la señal utilizada para habilitar las salidas de la SRAM, denominada como OE y definida como de tipo combinacional. Se indica también el pin de la tarjeta XS95 correspondiente.
Línea 13: Declaración de la señal utilizada para habilitar el chip SRAM, denominada como CE y definida como de tipo combinacional. Se indica también el pin de la tarjeta XS95 correspondiente.
Línea 14: Declaración de la señal utilizada para producir el reset del µC8031, denominada como RESET y definida como de tipo combinacional. Se indica también el pin de la tarjeta XS95 correspondiente.
Línea 15: Declaración de la señal utilizada como reloj por el µC8031, denominada como RELOJ y definida como de tipo combinacional. Se indica también el pin de la tarjeta XS95 correspondiente.
Línea 16:
Línea 17: Declaración de la señal asociada al bit más significativo del puerto paralelo del PC, denominada como PAR7 y que será utilizada para resetear el µC8031.
Línea 18: Declaración del Alias Bus_dir_baja_sram para el conjunto de señales [D7..D0].
Línea 19: Declaración del Alias P0_uc8031 para el conjunto de señales [A7..A0].
Línea 21..22: Sentencia que indica que las salidas de las señales de tipo registro [D7..D0] serán las señales definidas en el conjunto P0_uc8031, las que serán capturadas con los cantos de bajada de señal ALE.
Línea 23: Sentencia para la aplicación de la señal CLK a la señal RELOJ.
Línea 24: Sentencia para la aplicación de la señal A15 a la señal CE.
Línea 25: Sentencia para la aplicación del resultado de la operación AND entre las señales PSEN y RD a la señal OE.

Línea 26: Sentencia para el reset del uC8031 con la señal PAR7.
Línea 27:
Línea 28: Fin de la descripción para el módulo uc8031.

El programa escrito en C, puede emplear la SRAM como memoria estática externa para datos, cuidando que en la definiciones de las variables que deben ubicarse en la RAM estén en direcciones mayores que la zona empleada para almacenar el programa. Con la memoria RAM disponible en la tarjeta pueden direccionarse desde la dirección 0x0000 a la 0x7FFF. Podría utilizarse la zona de 0x0000 hasta 0x3FFF para programas, y la zona desde 0x4000 hasta 0x7FFF para Ram externa.
Por ejemplo para definir una zona de 256 bytes en RAM externa en la dirección 0x4000, puede emplearse:

at 0x4000 xdata unsigned char arreglo[256];

Debe notarse que esta forma de definir los segmentos es dependiente del compilador que se está empleando.

En el Laboratorio.

Grabar en el CPLD el programa para conectar la memoria, el reloj y la señal de reset.

```
MODULE uc8031
TITLE 'Módulo que implementa la lógica para el funcionamiento del uC8031'
 DECLARATIONS
 A15,A7..A0       PIN 34,35,36,37,39,40,41,43,44;
 PSEN   PIN 13;
 RD       PIN 32;
WR        PIN 63;
 CLK     PIN 9;
ALE      PIN 20;
D7..D0  PIN 2,83,3,1,84,82,79,75 istype 'reg';
OE        PIN 62 ISTYPE ‘COM’; // señal habilitación de salidas SRAM
CE        PIN 65 ISTYPE ‘COM’; // señal habilitación de chip SRAM
RESET  PIN 45 ISTYPE ‘COM’; // señal de reset del uC8031
RELOJ  PIN 10 ISTYPE ‘COM’; // señal de entrada de clk del uC8031
PAR7    PIN 80;  // Bit más significativo del puerto paralelo del PC
Bus_dir_baja_sram = [D7..D0];
P0_uc8031  = [A7..A0];
EQUATIONS
Bus_dir_baja_sram.CLK = ! ALE;
Bus_dir_baja_sram := P0_uc8031;
RELOJ = CLK;
CE = A15;
OE = PSEN & RD;
RESET = PAR7; // bit más significativo de la puerta paralela resetea al uC
END uC8031      //---- fin descripción para lógica del uC8031
```

Compilar y generar el código hexadecimal, formato Intel, del siguiente programa en C.

```
#include <reg51.h>
void main (void)
{     for(; ; ) { P1= 0x0; P1 = 0xFF;}  }
```

Cargar mediante XSLOAD el programa en la RAM.

- Observar un ciclo de lectura de la instrucción. Ciclo de búsqueda(fetch).

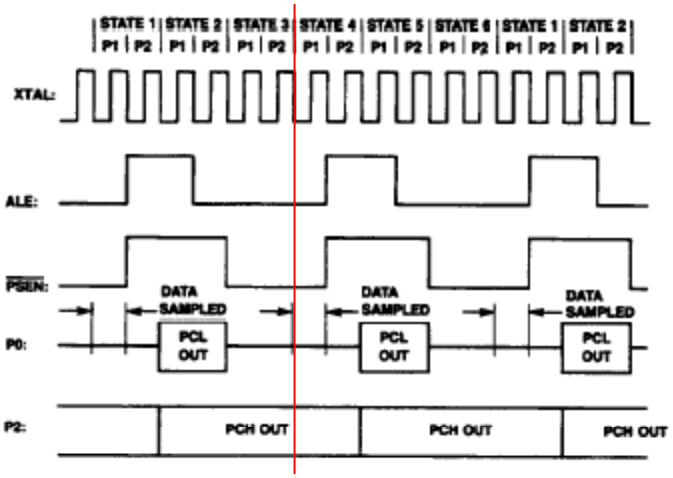
Observar mediante el osciloscopio los ciclos de ALE y PSEN al ejecutar repetidamente las siguientes instrucciones(que son generadas por el programa anterior).

```
start:    mov   P1,#00
          mov   P1,#255
          jmp   start
```

Puede agregar al código en el CPLD un pequeño programa que determine cuando se efectúa el fetch de una de las instrucciones, para generar una señal de trigger para el osciloscopio. Esto puede hacerse decodificando el bus de datos o el bus de direcciones en instantes en que las señales de control toman determinados valores.

Notar que cuando ALE tiene un canto de bajada y está alta PSEN', el bus de direcciones contiene la dirección de la instrucción. Ver fig. 36 en página 3-34 del manual del usuario.

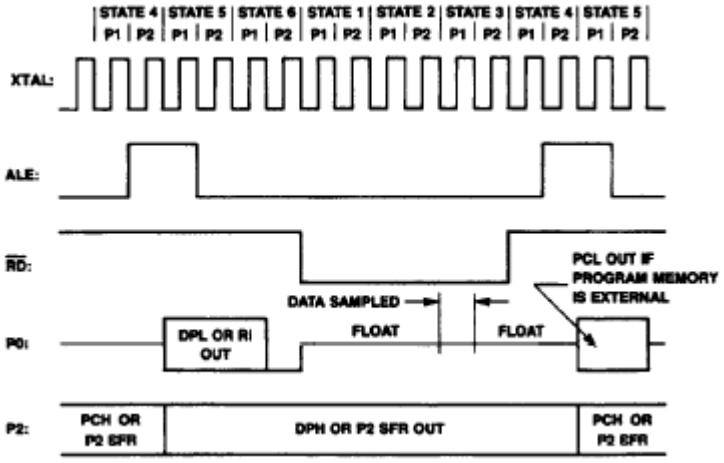
El siguiente diagrama muestra las relaciones entre las señales en un ciclo de búsqueda de instrucción.

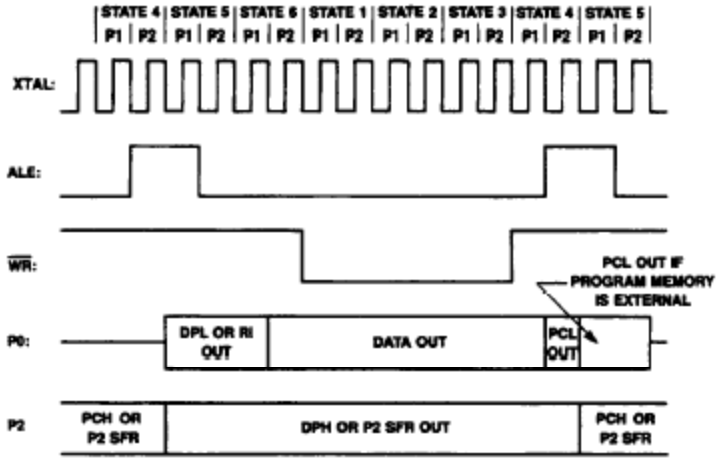


- Observar un ciclo de lectura y escritura de la ram externa.

Modificar los programas para escribir y leer un byte de la ram externa, visualizando en el osciloscopio las señales WE' y OE', ALE y PSEN.

Los diagramas siguientes muestran las relaciones entre las señales en un ciclo de lectura y escritura a la ram de datos externa.


STATE 4 P1 P2
STATE 5 P1 P2
STATE 6 P1 P2
STATE 1 P1 P2
STATE 2 P1 P2
STATE 3 P1 P2
STATE 4 P1 P2
STATE 5 P1 P2
XTAL:
ALE:
RD:
PCL OUT IF PROGRAM MEMORY IS EXTERNAL
DATA SAMPLED
FLOAT
FLOAT
P0:
DPL OR RI OUT
P2:
PCH OR P2 SFR
DPH OR P2 SFR OUT
PCH OR P2 SFR


STATE 4 P1 P2
STATE 5 P1 P2
STATE 6 P1 P2
STATE 1 P1 P2
STATE 2 P1 P2
STATE 3 P1 P2
STATE 4 P1 P2
STATE 5 P1 P2
XTAL:
ALE:
WR:
PCL OUT IF PROGRAM MEMORY IS EXTERNAL
P0:
DPL OR RI OUT
DATA OUT
PCL OUT
P2
PCH OR P2 SFR
DPH OR P2 SFR OUT
PCH OR P2 SFR